Amendments to the Specification:

Please amend the following paragraphs as indicated:

[0007] If desired, a plurality of arrays 400, together with related X decoders, Y decoders, program/verified circuitry, data registers, and the like are provided, for example as taught by U.S. Patent 5,890,192, issued March 30, 1999, and assigned to SanDisk Corporation, the assignee of this application, which is hereby incorporated by this reference. Related memory system features are described in U.S.~~co-pending~~ patent ~~application serial~~ no. ~~09/505,555~~ 6,426,893~~, filed February 17, 2000 by Kevin Conley et al.~~, which ~~application~~ is expressly incorporated herein by this reference.

[0015] In some prior art systems having large capacity memory cell blocks that are divided into multiple pages, the data from a block that is not being updated needs to be copied from the original block to a new block that also contains the new, updated data being written by the host. In other prior art systems, flags are recorded with the user data in pages and are used to indicate that pages of data in the original block that are being superceded by the newly written data are invalid. A mechanism by which data that partially supercedes data stored in an existing block can be written without either copying unchanged data from the existing block or programming flags to pages that have been previously programmed is described in U.S.~~co-pending~~ patent ~~application "Partial Block Data Programming and Reading Operations in a Non-Volatile Memory", serial~~ no. 6,763,424 ~~09/766,436, filed January 19, 2001 by Kevin Conley~~, which ~~application~~ is expressly incorporated herein by this reference.

[0017] Such non-volatile memory systems include one or more arrays of floating-gate memory cells and a system controller. The controller manages communication with the host system and operation of the memory cell array to store and retrieve user data. The memory cells are grouped together into blocks of cells, a block of cells being the smallest grouping of cells that are simultaneously erasable. Prior to writing data into one or more blocks of cells, those blocks of cells are erased. User data are typically transferred between the host and memory array in sectors. A sector of user data can be any amount that is convenient to handle, preferably less than the capacity of the memory block, often being equal to the standard disk drive sector size, 512 bytes. In one commercial architecture, the memory system block is sized to store one sector of user data plus overhead data, the overhead data including information such as an error correction code (ECC) for the user data stored in the block, a history of use of the block, defects and other physical information of the memory cell block. Various implementations of this type of non-volatile memory system are described in the following United States patents and pending applications assigned to SanDisk Corporation, each of which is incorporated herein in its entirety by this reference: Patents nos. 5,172,338, 5,602,987, 5,315,541, 5,200,959, 5,270,979, 5,428,621, 5,663,901, 5,532,962, 5,430,859, ~~and~~ 5,712,180, ~~and application serial nos. 08/910,947~~6,333,762, ~~filed August 7, 1997,~~ and 6,151,248 ~~09/343,328, filed June 30, 1999~~. Another type of non-volatile memory system utilizes a larger memory cell block size that stores multiple sectors of user data.

[0019] Since the programming of data into floating-gate memory cells can take significant amounts of time, a large number of memory cells in a row are typically programmed at the same time. But increases in this parallelism cause increased power requirements and potential disturbances of charges of adjacent cells or interaction between them. United States patent no. 5,890,192 of SanDisk Corporation, which is incorporated above, describes a system that minimizes these effects by simultaneously programming multiple pages (referred to as chunks in that patent) of data into different blocks of cells located in different operational memory cell units (sub-arrays). Memory systems capable of programming multiple pages in parallel into multiple sub-array units are described in U.S. ~~co-pending~~ patent ~~applications serial~~ no. ~~09/505,555~~ 6,426,893~~, filed February 17, 2000 by Kevin Conley et al.~~, which is incorporated by reference above, and U.S. patent ~~serial~~ no. ~~09/703,083~~ 6,570,785~~, filed October 31, 2000, by John Mangan et al.~~, which ~~application~~ is expressly incorporated herein by this reference.

[0035] In order to improve performance by reducing programming time, a goal is to program as many cells in parallel as can reasonably be done without incurring other penalties. One implementation divides the memory array into largely independent sub-arrays or units, each unit in turn being divided into a large number of blocks, as described in U.S. patent ~~applications serial~~ no. ~~09/505,555~~ 6,426,893 ~~, filed February 17, 2000, by Kevin Conley et al.~~ and ~~serial~~ no. ~~09/703,083~~ 6,570,785~~, filed October 31, 2000, by John Mangan et al.~~, which are incorporated by reference above. Pages of data are then programmed at the same time into more than one of the units. Another configuration further combines one or more of these units from multiple memory chips. These multiple chips may be connected to a single bus (as shown in Figure 2) or multiple independent busses for higher data throughput.

[0038] Figure 3 is a block diagram showing some elements of a non-volatile memory such as that in Figures 1 and 2. The other elements are suppressed in Figure 3 in order to simplify the discussion, but are shown in more detail in, for example, U.S. patents ~~patent applications serial~~ nos. ~~09/505,555~~ 6,426,893 and ~~09/703,083~~ 6,570,785, incorporated by reference above.

[0040] Each of the memory units 131-i has an array of memory cells MEM 133-i in which the data is stored and a register REG 135-i for storing temporarily storing the data as it is transferred between the array 133-i and bus 121. Each of the arrays is shown subdivided into, here, four subarrays into which pages may be programmed in parallel as described in U.S. patents ~~patent applications serial~~ nos. ~~09/505,555~~ 6,426,893 and ~~09/766,436~~ 6,763,424, incorporated by reference above. The controller 101 and the memory units 131-i are commonly placed on separate chips and may be referred in that manner below, although one or more of the memory units can be on the same chip as the controller. Alternately, more than one of the memory units may be formed on the same integrated circuit, but on a distinct chip from the controller 101. The memory units 131-i and controller 101 may form a single card for attachment to a host, may all be embedded in the host, or just the controller 101 may be embedded in the host with the memory units 131-i forming a removable card structure. In any of the embodiments, each of the memory arrays 131 are independent in that the controller can independently load command, address, and transfer data to them.

[0046] Parallel page programming on a single chip or memory unit is shown in Figure 5a, again for the four-sector example. The first set of data is loaded via the external interface from the host into buffer A as before, but now after time t_1 the data for all four sectors are transferred to the memory prior to the beginning of programming. Although this is shown as a transfer of data for sector 1 followed by sector 2 and so on, more generally portions of each are transferred until complete as described in U.S. patents ~~patent applications serial~~ nos. ~~09/505,555~~ 6,426,893 and ~~09/766,436~~ 6,763,424, incorporated by reference above. Once the data transfers for all the pages are complete in their respective data registers, the four pages are programmed in parallel into their respective memory cells until verified at a time t_2. During the interval between t_1 and t_2, the data for the next four pages can be transferred from the host into buffer B. After time t_2 this second set of data can then be transferred and programmed in the same or another memory unit and so on.

[0057] As already noted, in Figures 6a and 7, the sizes of the blocks are just meant to be illustrative of the causal relationships and may not accurately reflect block sizes in actual memories. Concerning the relative size of the various time intervals involved, in a particular embodiment exemplary values are ~120μs to transfer four pages of data from the host into a buffer, ~160μs to transfer this data set from the buffer to a memory unit's register, ~ 200μs to write the four pages, and ~1-4ms for the erase time. Although the process of Figure 7 will be faster than that of Figure 6a, it may be less reliable in some circumstances since it does not maintain the data in the buffers until its successful programming is confirmed. In a set of alternate embodiments the data may be maintained elsewhere, allowing the buffer to be reloaded after transfer as in Figure 7 for increased speed while keeping an uncorrupted copy of the data set. For example, it could be maintained in the controller, although this increases the amount of RAM required in the controller. In another embodiment, it is maintained on the memory unit itself, thereby saving the need to re-transfer the data set should it be needed. Referring to Figure 3, each memory unit 131-i would have additional RAM memory, for example by enlarging register 135-I, where a back-up copy of the data set could be loaded in at the same time it is transferred to register REGi 135-i. Such an arrangement is described in ~~copending~~ U. S. patent ~~application serial~~ number ~~09/751,178, filed December 28, 2000~~6,349,056. Also as described there, this arrangement also lets the end result of the programming process be verified without transferring the result back to the controller to be checked with error correction code.

[0059] Also as noted above, the embodiments of Figures 6 and 7 readily extend to more than two buffers, more than two memory units, or both. For example, referring to Figure 6b, data could be loaded into a third buffer after time t_2, then transferred and written into a third memory unit following the transfer in interval 62b. Additionally, these embodiments can be combined with the sort of metablock operation described in U.S. patent ~~application serial~~ no. ~~09/766,436~~6,763,424, which is incorporated by reference above, where blocks from different units can be operated together as a metablock.